Lamourre Étienne

Réalisation d'un site personnel

Partie 3 : Cours php et mySQL

Constitution du rapport

Le rapport est constitué de plusieurs parties séparées qui contiennent respectivement :

- ***Partie 1*** : Une description externe du genre «manuel d'utilisateur» avec la carte de navigation dans la partie visible du site. Cette partie doit être accessible à tous dans le site lui-même.
- ***Partie 2*** : Une description interne décrivant l'organisation des fichiers et le fonctionnement des programmes. Pour cette partie en particulier, on insiste sur le fait que tout schéma doit être accompagné d'explications. Le «format» du document résultant est laissé libre : ce peut être un document HTML ou un document pdf. Cette partie doit être contenue dans la partie privée.
- ***Partie 3*** : Une partie cours décrivant le fonctionnement de sites web dynamiques avec php et mySQL, d'un point de vue général. Elle doit aussi contenir la description soignée d'un point de code, comme par exemple le code qui calcule le menu horizontal ou adapte le «centre». Cette partie peut-être dans la partie publique du site, et dans un fichier pdf.
- ***Partie 4*** : Une partie «meta» décrivant le déroulement et l'organisation du travail, ainsi que les logiciels utilisés. Avec un point de vue plus ambitieux, cette partie correspond aux documents rédigés dans le cadre de la gestion de projet. Cette partie doit être dans la partie privée du site, dans un fichier pdf.

I - Fonctionnement du Php

PHP est un langage de programmation orienté site web qui s'intègre parfaitement dans les pages HTML et CSS. Il est compatible avec la base de données MySQL ce qui permet d'élaborer des applications rapides, avec une actualisation en temps réel des données.

a - Architecture client / serveur

Le navigateur internet est le client. C'est lui qui demande la page web au serveur. Le client se fournit dans la base de donnée de fonctions et de contenus du serveur. PHP et MySQL sont des langages SERVEUR.

Description d'une requête HTTP basique :

- Le navigateur envoi au serveur, qui héberge le site, l'adresse que l'on a tapé dans la barre d'adresses du navigateur.
- Le serveur retourne après traitement le contenu de la page au navigateur.

Observation :

Avec ou sans php, le serveur traite la demande et retourne la page. Additionner du php au traitement n'a aucune incidence sur le parcours basique de la requête http. Le serveur va se charger de traiter les scripts php et se chargera aussi du traitement des données des tables SQL, avant de renvoyer la page terminée au navigateur.

b - Coté serveur

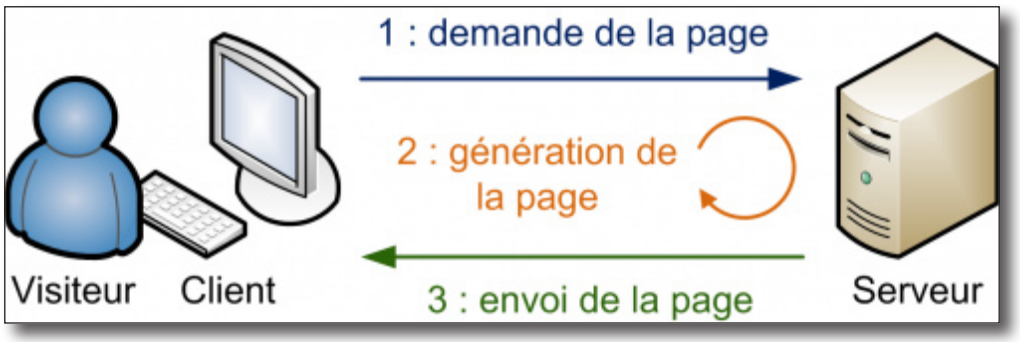
Le php est interprété directement et uniquement par le serveur. La seule manière d'accéder aux pages php est par un logiciel de FTP (fetch, cyber duck ...).

Le serveur ne renvoi que des données (le résultat des traitements de requêtes ou de fonctions..) qui constituent le source de la page.

Les différents langages comme le HTML, JavaScript ou CSS sont des langages clients. Les scripts de ces différents langages sont télécharger sur les ordinateurs et ensuite interprétés par le navigateur.

Voici les étapes de chargement d'une page lorsque le site est dynamique :

- Le client demande au serveur à voir une page web.
- Le serveur prépare la page spécialement pour le client.
- Le serveur lui envoie la page qu'il vient de générer.



La page web est générée à chaque fois qu'un client la réclame. C'est précisément ce qui rend les sites dynamiques vivants, le contenu d'une même page peut changer d'un instant à l'autre.

II - Fonctionnement de MySQL

MySQL est un système de gestion de base de données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server.

Utilisation

Une base de données permet d'enregistrer des données de façon organisée et hiérarchisée.

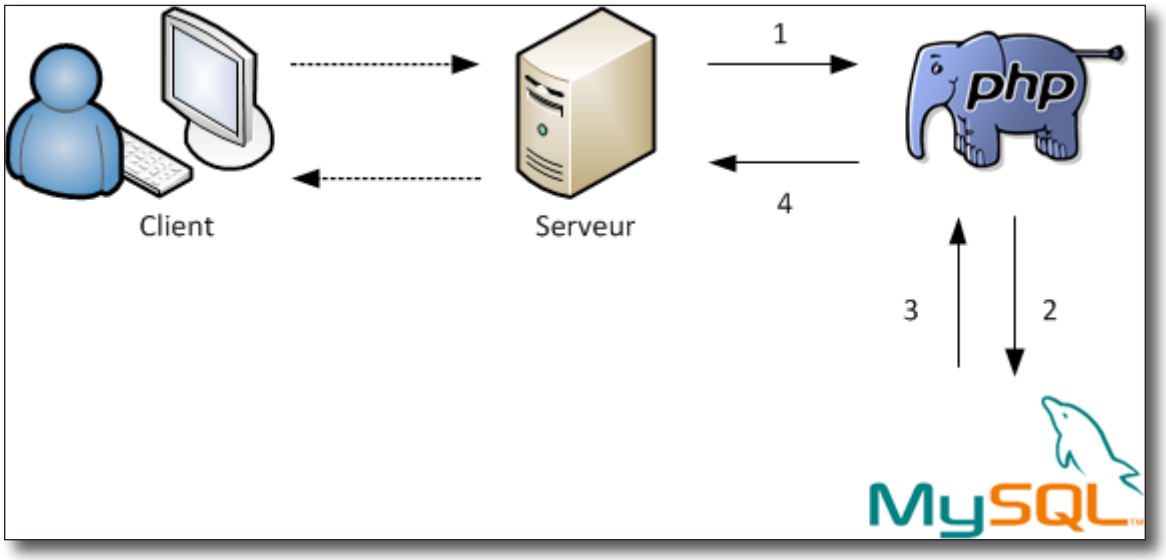
MySQL fait partie du quatuor LAMP : Linux, Apache, MySQL, PHP. Il appartient également à ses variantes WAMP (Windows) et MAMP (Mac).

Le couple PHP/MySQL est très utilisé par les sites Web et proposé par la majorité des hébergeurs Web. Plus de la moitié des sites Web fonctionnent sous Apache, qui est le plus souvent utilisé conjointement avec PHP et MySQL.

Wikipédia utilise MySQL, ainsi que de nombreuses entreprises (Google, Yahoo!, YouTube, Adobe,...).

PHP fait la jonction entre nous et MySQL :

On ne va pas pouvoir parler à MySQL directement. Seul PHP peut le faire. C'est donc PHP qui va faire l'intermédiaire entre nous et MySQL. On devra demander à PHP : «Va dire à MySQL de faire ceci.».



Voici ce qu'il peut se passer lorsque le serveur a reçu une demande d'un client :

1. Le serveur utilise toujours PHP, il lui fait donc passer le message.
2. PHP effectue les actions demandées et se rend compte qu'il a besoin de MySQL. En effet, le code PHP contient à un endroit «Va demander à MySQL d'effectuer cette tâche». Il fait donc passer le travail à MySQL.
3. MySQL fait le travail que PHP lui avait soumis
4. PHP renvoie au serveur que MySQL a bien fait ce qui lui était demandé.

III - Description d'un point de code

Dans mon site personnel figure une zone d'identification permettant de sécuriser certains dossiers par un mot de passe associé à un login. Dans cette partie, je détaillerais le code php utilisé pour réaliser l'authentification des utilisateurs.
Ce script fonctionne grâce à deux fichiers distincts : un fichier log.php et un fichier login.php.

a - Le fichier log.php

Voici le code contenu dans le fichier log.php :

```php
<?php
if (($Utilisateur == "axeo") && ($Mdp == "axeo")) //Login et Password
{ header ("Location: /_axeo"); // la page affichée une fois que l'utilisateur aura rentrer le bon login et bon password
 exit;
}


if (($Utilisateur == "pagora") && ($Mdp == "licpro2010")) //Login et Password
{ header ("Location: /_rapports"); // la page affichée une fois que l'utilisateur aura rentrer le bon login et bon password
 exit;
}


else
{ header ("Location: login.php3?Message=Invalide");
 exit;
}
?>
```

Ce fichier contient les logins et mots de passe des utilisateurs.

- Le premier paragraphe : si l'utilisateur est *axeo* et son mot de passe est *axeo*, j'affiche le dossier *_axeo*.
- Le deuxième paragraphe : si l'utilisateur est *pagora* et son mot de passe est *licpro2010*, j'affiche le dossier *_rapports*.
- Le troisième paragraphe : sinon (si l'utilisateur n'est ni *pagora* ni *axeo* et que le mot de passe est erroné), j'affiche le message «invalide» (message défini dans le fichier *login.php*.

b - Le fichier login.php

Le lien identification du menu horizontal renvoi sur le fichier *ident.php*. Ce fichier est composé de toute les informations nécessaires au bon affichage de la page (style css, favicon, etc). Afin d'utiliser le script php, j'ai utilisé la fonction include qui appelle directement le fichier login.php qui contient uniquement le script :

```php
<?php
    include ('login.php')
?>
```

Voici le code contenu dans le fichier login.php :

```php
<?php

if ($Message == "Invalide")
{ print ("<B><CENTER><FONT COLOR=RED>Le Login et Password ne sont pas valide !</FONT></CENTER></B>\n");
}
print ("<FORM ACTION=\"/log.php\" METHOD=POST>\n");
print ("Login :  <INPUT TYPE=TEXT NAME=Utilisateur SIZE=31><BR><BR>\n");
print ("Password : <INPUT TYPE=PASSWORD NAME=Mdp SIZE=27><BR><BR>\n");
print ("<INPUT TYPE=SUBMIT NAME=SUBMIT VALUE=\"Go\">\n");
?>
```

Ce fichier contient les champs de saisie des logins et mots de passe.

- Le premier paragraphe : on commence par vérifier que les informations saisies soient correctes. Si le message revient «invalide», on affiche «Le login et Password ne sont pas valide !».
- Le deuxième paragraphe : on créé un champ de saisie pour le login.
- Le troisième paragraphe : on créé un champ de saisie pour le mot de passe.
- Le quatrième paragraphe : on créé un bouton «go» pour valider la saisie.